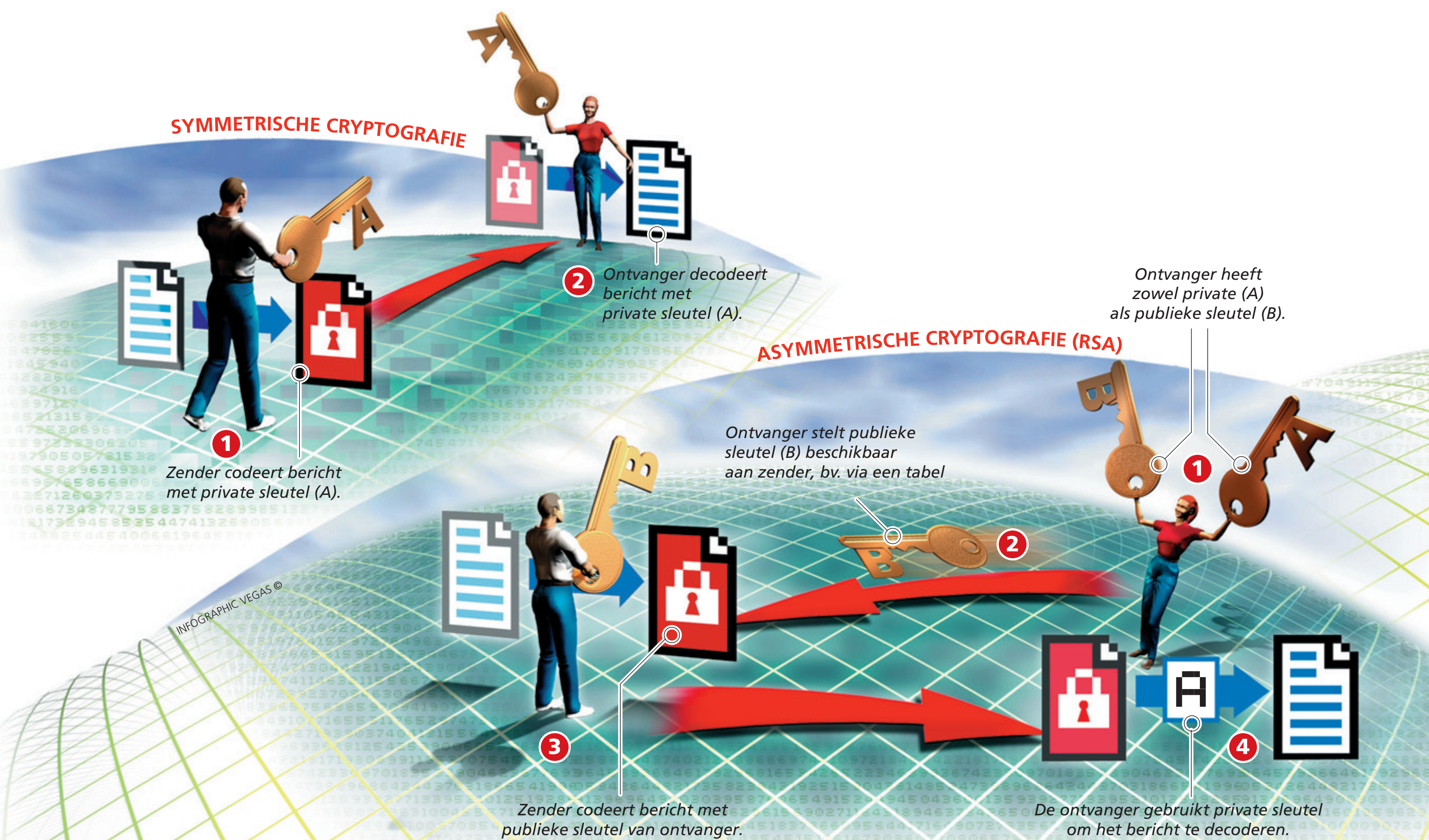
Beveilig je gegevens door ze te encrypteren

Spelen met sleutels

Soms is het handig mails of gegevens te beveiligen tegen al te nieuwsgierige aagjes. Tegenwoordig kan dit door je gegevens te versleutelen volgens een bepaald patroon, ook wel encryptie genoemd. Het ontsleutelen van die gegevens is dan weer decryptie. Hoe dat allemaal in elkaar zit en wat het nut ervan is, leggen we je hier uit.

Het woord cryptografie vindt zijn oorsprong in het Griekse kryptós lógos, wat zoveel betekent als 'verborgen woord'. Het voordeel van een gecodeerde boodschap is vrij voor de hand liggend. Je stuurt een mailtje met vertrouwelijke informatie naar een kennis. Die mail wordt onderschept door iemand met slechte bedoelingen. Twee dagen later is je bankrekening geplunderd. Je hoeft het zelfs niet zo ver te zoeken. Op je harde schijf staan waarschijnlijk wel een aantal persoonlijke documenten, waarvan je liever niet hebt dat anderen ze te zien krijgen. Wat je kan doen om dat te vermijden, is die documenten coderen. Cryptografie beperkt zich overigens niet tot het domein der computers. Ook bij een telefoontje kan encryptie gebruikt worden. Dat noemen we Voice Encryption en vermijdt alvast dat iemand het telefoontje naar je minnares afluistert. Betaaltelevisie werkt even-



eens met encryptie. Iedereen ontvangt thuis het signaal van Canal+. Wie niet betaalt voor een abonnement, krijgt enkel een gestoord beeld te zien. Abonnees hebben een decoder in hun bezit. Die decoder is in staat om het inkomende signaal te decoderen of decrypteren.

Hoe werkt encryptie nu eigenlijk? Een eenvoudig voorbeeld is een bericht coderen door alle letters enkele posities in het alfabet op te schuiven. Daarbij verhoog je elke letter met bv. twee posities. Je maakt van een A een C, van een Z een B enzovoorts. Een eenvoudige zin als 'Hoe gaat het' wordt 'Jqg iccv jgv', wat totaal onleesbaar is voor iemand die de sleutel niet kent. Met de sleutel bedoelen we het stukje informatie dat nodig is om de boodschap te (de)coderen. In dit geval is dat letter+2. Het nadeel aan deze vorm van cryptografie is dat beide partijen overeen moeten komen welke sleutel gebruikt wordt. Dat houdt in dat op een bepaald moment een vorm van communicatie nodig is, waarbij de sleutelwaarde doorgegeven wordt. Dat kan via de telefoon zijn, via een mail of via een FedEx-pakje. Tijdens die overdracht kan de sleutel onderschept worden. Dat is een beetje de Catch-22 van de cryptografie. Immers, vooraleer we een versleuteld bericht kunnen verzenden, moeten we een beveiligd kanaal met de ontvanger van dat bericht opzetten. Maar de bedoeling van die sleutel is nu net het opzetten van zo'n beveiligd kanaal. Die vorm van cryptografie noemen we cryptografie met een private sleutel, ofwel symmetrische cryptografie. Symmetrisch omdat zowel bij het coderen als het decoderen dezelfde sleutel gebruikt wordt.



Iedereen een sleutel

Nu is er wel een tweede manier, die de uitwisseling van een sleutel overbodig maakt. Dat is cryptografie met behulp van een publieke sleutel. Die techniek houdt in dat elke gebruiker twee sleutels in zijn bezit heeft. Een publieke en een private sleutel. De publieke sleutel wordt vrij verspreid en wordt gebruikt om boodschappen te coderen. De private sleutel is voor elke gebruiker persoonlijk, en dient om de boodschappen te ontcijferen. Alle verzonden data maken gebruik van publieke sleutels. Op geen enkel moment is het nodig dat een private sleutel uitgewisseld wordt. Ergens moet er een tabel aanwezig zijn, waarin alle publieke sleutels én de namen van hun eigenaars opgeslagen zitten. De enige vereiste is dat deze correlatie – tussen de publieke sleutels en hun eigenaars – opgesteld wordt door een te vertrouwen instantie. Uiteraard mag de private sleutel niet in de verkeerde handen terechtkomen. Verwarrend? Een concreet voorbeeldje: Jan wil communiceren met Anna. Jan zoekt in de – beveiligde – tabel naar de publieke sleutel van Anna. Hij codeert zijn bericht met behulp van Anna's sleutel en verzendt het bericht. Anna ontvangt het bericht dat gecodeerd is met haar publieke sleutel. Vermits zij de enige is die de corresponderende private sleutel heeft, is zij de enige die het bericht kan decoderen.

Hakken maar

Cryptografie omvat meer dan het coderen van tekstboodschappen. We moeten ook zeker zijn dat de ontvangen boodschap intact is én dat de boodschap afkomstig is van de rechtmatige zender. Met andere woorden, we willen een of andere vorm van echtverklaring. Daarvoor gebruiken we een digitale handtekening. Zo'n handtekening is gebaseerd op het originele document én de private sleutel van de verzender. Doorgaans wordt een digitale handtekening gemaakt door middel van hashing (verhakken). Een hash-waarde is een nummer dat gegenereerd wordt uit een uittreksel van de originele tekst. Eigenschappen van een hash-functie zijn dat het vrijwel onmogelijk is dat twee verschillende berichten dezelfde hash-waarde produceren, én dat het heel erg moeilijk is om vanuit de hash-waarde de originele tekst te berekenen. Na het berekenen van de hash-waarde wordt die waarde gecodeerd met je private sleutel. Merk op dat we hier enkel het uittreksel van de tekst coderen. Dat is omdat je oorspronkelijke bericht wel eens erg lang zou kunnen zijn, en encryptie met publieke sleutels neemt veel tijd in beslag. Verder wordt de integriteit van het bericht verzekerd door niet het volledige bericht, maar enkel het uittreksel te voorzien van een digitale handtekening. De gecrypteerde hash-waarde samen met het hash-algoritme noemen we de digitale handtekening.
Allemaal goed en wel, maar waar is dat nu goed voor? Wel, stel dat Anna een bericht wil zenden naar Jan. Anna maakt een hash-waarde van haar bericht en codeert die hash-waarde. Die gecodeerde hash-waarde bevat tevens de informatie over het hashing-algoritme. Vervolgens verzendt ze haar bericht, dat ze naar eigen keuze wél of niet kan coderen, samen met de gecodeerde hash-waarde. Jan decodeert – indien nodig - het bericht én de hash-waarde. Vervolgens maakt Jan zélf een hash-waarde van het gedecodeerde bericht. Dat doet hij door gebruik te maken van het

algoritme dat hij van Anna heeft ontvangen. Jan vergelijkt zijn hash-waarde met de waarde van Anna. Indien beide waarden gelijk zijn, kan hij met grote zekerheid zeggen dat het bericht intact aangekomen is.

DES en RSA

Nee, DES en RSA zijn geen enge ziektes, maar twee van de meest gebruikte encryptietechnieken. DES staat voor Data Encryption Standard en is de bekendste en meest gebruikte symmetrische encryptietechniek. Daarbij wordt gebruik gemaakt van een reeks van ingewikkelde rekenkundige bewerkingen. Wie wil weten hoe dit in elkaar zit, kan zich verdiepen in de precieze uitleg op [**www. tropsoft.com/strongenc/des.htm**]. Hou wel dat wiskundig naslagwerk bij de hand. RSA encrypteert met een publieke sleutel. De methode is genoemd naar haar drie uitvinders, Rivest, Shamir en Adleman. Bij RSA zijn de private en de publieke sleutel op een rekenkundige manier aan elkaar gelinkt. De beveiliging werkt volgens het zogenaamde 'valdeur-principe'. Dat wil zeggen, encryptie is heel eenvoudig uit te voeren, en de decryptie is zo goed als onmogelijk, tenzij je de private sleutel hebt. Om de code te kraken – dus om de private sleutel af te leiden uit de publieke – moet je de publieke sleutel met een gigantisch groot cijfer vermenigvuldigen. Daarvoor is heel veel rekenkracht en heel veel tijd nodig.
Hoe veilig een bepaalde encryptering is, kan je dan ook afleiden uit de lengte van de sleutel. Die wordt uitgedrukt in bits. Zo gebruikt DES een 64-bits sleutel om de data te encrypteren. Van die 64 bits worden er acht gebruikt als 'parity bits', die controleren of de eerste 56 bits wel correct zijn doorgezonden. De overige 56 bits worden gebruikt om te encrypteren. Daarmee kunnen ongeveer 7,2 x 10^{16} mogelijke sleutelwaarden gevormd worden. Een computer die tien miljoen sleutelwaarden per seconde zou kunnen analyseren, heeft nog steeds méér dan 228 jaar nodig om alle mogelijke combinaties te overlopen. De kracht van deze encryptiemethodes heeft ertoe geleid dat de Amerikaanse overheid de export van software die gebruik maakt van symmetrische sleutels langer dan 40 bits heeft verboden. Anders bestaat de kans dat ze onderschepte berichten zélf niet meer kunnen ontcijferen. Op [**http://world.std.com/~franl/crypto/rsa-example.html**] kan je een voorbeeldje van RSA-encryptie terugvinden. DES is trouwens veel sneller dan RSA, dat door het gebruik van twee sleutels meer berekeningen vergt.
Wil je nu zelf een berichtje coderen, dan kan je gebruik maken van PGP (Pretty Good Privacy). Die techniek is gemakkelijk te gebruiken én het programma is gratis te downloaden op de website van PGP [**www.pgp.com/products/freeware**].
Zo, hiermee heb je alvast de basis van de cryptografie onder de knie. Waar wacht je nu nog op om een mooie liefdesbrief naar je vriend(in) te sturen ? In gecodeerde vorm uiteraard. Als hij/zij dat niet romantisch vindt, dan weten wij het ook niet meer...

— Benjamin Carlier —

BEVEILIGING OP HET INTERNET

Ook SSL is een heel belangrijke toepassing van encryptie. SSL staat voor Secure Sockets Layer, een protocol ontwikkeld door Netscape. Het SSL-protocol is dat kleine icoontje in de vorm van een slot dat onderaan op je internetbalk verschijnt wanneer je op een beveiligde pagina surft. Dat protocol zorgt voor die beveiliging door gebruik te maken van encryptietechnieken en digitale handtekeningen. SSL gebruikt een combinatie van de twee technieken, in de zin dat publieke-sleutel-encryptie gebruikt wordt om een private sleutel te verzenden. In de praktijk komt het erop neer dat SSL de veiligheid van een bepaalde verbinding onderzoekt, en dat vóór de (vertrouwelijke) data verzonden wordt. Die verificatie noemen we het SSL Handshake Protocol. Meer informatie, inclusief de werking van die handdruk, kan je vinden op de website van Netscape [**http://wp.netscape.com/eng/ssl3/index.html**].

